Archives
S
89
.E21
no.493

A Modified General Simplex Method For Solving Linear Programming Problems

By
BENNETT B. FOSTER
and
RICHARD R. WEYRICK

Station Bulletin 493

Agricultural Experiment Station
University of New Hampshire
Durham, New Hampshire

SHOUT
SHARE
IT

Table of Contents

Introduction 1

An Example Problem 2
Technique for solution, p. 3
Rules for the pivoting procedure, p. 4

Minimizing Problems and Problems where Zero Production is not Allowed 7

Summary of Rules for Determining Pivot Row and Pivot Column and the Pivoting Operation 10

Equality Constraints and No Solution Situations 12
Equality constraint, p. 12
"No solution" situations, p. 12

Computer Program 13

Selected References 13

Appendix A. Fortran II Program with Instructions for Data Cards 15

Appendix B. Derivation of Pivoting Rules and Pivot Row and Column Selection Rules 19

February 1968

The research project contributing the technique described in this bulletin is funded through the McIntire - Stennis Cooperative Forestry Research Act of 1962. The authors are assistant professors in the Department of Forest Resources.

Introduction

Most formal explanations and many computer programs for solving linear programming problems follow the simplex "tableau" method. As desirable as this method is for enabling the beginning student to see the "whys" and "hows" of linear programming, it is rather complicated. If followed in writing an L. P. problem solving program where the origin (zero production) is not feasible, this method unnecessarily uses up a sometimes short supply of computer storage space and is vulnerable to a computer "bug" resulting from an inadequate value "M" for the slack variable(s).

This bulletin describes a technique (or algorithm) for solving L. P. problems that is based on rules derived by simple high school algebra rather than the intuitive descriptive approach or the more formal mathematical approach. It draws on the simple mathematical characteristics of the derived rules to determine the logical sequences of the problem solution and also to eliminate the need for artificial variables and high negative "M" values in problems where zero production is not allowed.

The technique is such that, for a clearer understanding, the explanation will begin with the set-up of a simple problem and proceed rapidly to the rules of the technique (the algorithm). It is important to point out that the *rules* presented herein are specifically oriented to the method presented for setting up the problem constraints.

The precise point of departure of this technique from those presented by others (Baumol [1], Heady and Candler [2], Stiefel [3], Vajda [4, 5] for example) is the placing of the equal sign when the inequality constraints are changed into equalities.[1] This modification in turn leads to a unique method of handling "greater-than-or-equal-to" constraints that does not require the use of artificial variables and the corresponding high negative values.

[1] $S_i = \sum X_i \pm C_i$ rather than $\sum X_i \pm S_i = C_i$

An Example Problem

Given: the following inequality constraints:[1]

$$\begin{aligned} 2X_1 + 3X_2 &\leq 24 \\ .5X_1 + .25X_2 &\leq 3 \\ X_1 &\leq 5 \end{aligned}$$

and objective function:

$$Z = 3X_1 + 2X_2 \quad \text{(max.)}$$

The inequality constraints are then changed into equalities by adding or subtracting slack variables.

$$\begin{aligned} 2X_1 + 3X_2 + S_1 &= 24 \\ .5X_1 + .25X_2 + S_2 &= 3 \\ X_1 + S_3 &= 5 \end{aligned}$$

In order to fit these equality constraints into this modified simplex technique, they are restated in terms of the positive slack variables, and set up in matrix form.

$$\begin{aligned} S_1 &= -2X_1 - 3X_2 + 24 \\ S_2 &= -.5X_1 - .25X_2 + 3 \\ S_3 &= -X_1 + 5 \end{aligned}$$

Restated Constraints

	X_1	X_2	1(Po)
$S_1 =$	-2	-3	24
$S_2 =$	$-.5$	$-.25$	3
$S_3 =$	-1	0	5
$Z =$	3	2	0

Constraint Matrix

The box in the lower right-hand corner of the matrix is the *objective solution.* The initial solution is zero because, as the table is read, there are 24 units of S_1, 3 units of S_2, and 5 units of S_3 in the solution. In other words, none of the resources have yet been used; therefore, the objective solution equals zero. However, the first two figures in the bottom row (objective function) indicate that the objective solution can be increased by three for each unit of X_1 brought into solution and by 2 for each unit of X_2 brought into solution.

[1] The other constraints generally assumed in LP problems are that the values X_1, X_2, S_1, S_2, etc., are non-negative; that is, the solution cannot contain negative amounts of an activity. These constraints are assumed in the algorithm presented, and are explicitly stated in the requirements from which the rules are developed (see p. 22, requirement A).

The Technique for Solution

In the following explanation each figure in the matrix will be called an "element."

The problem is solved by attempting to bring X_1 and X_2 (the activities) into solution. The resources (S_i) will be used to produce the activities (X_i). This will be done by mathematically exchanging rows and columns of the matrix, a row and a column at a time, in a series of matrix-forming processes called "iterations."

The process of exchanging a row and a column is called "pivoting" and is accomplished by using a set of rules described on page 4 and derived in Appendix B. The row and column that are exchanged in a pivoting process are known as "pivot row" and "pivot column," and the common element is known as the "pivot element." Under *no* circumstance will the *bottom row* or *last column* be considered as the pivot row or pivot column.

Pivot Column. With the exception of the last column, any column with a positive bottom element can be used as the pivot column,[1] as long as it contains a potential pivot element (see below).[2] The procedure outlined in this bulletin specifies the pivot column as the one with the *largest* positive bottom element, but actually the choice is arbitrary. In the example problem the pivot column will be the X_1 column.

Pivot Row. The appropriate row or "pivot row" is determined by the resource that is most limiting in the production of the pivot column activity. This of course excludes the bottom or objective function row. According to the constraints of the example problem (and also the matrix table if signs are ignored), the available supply of S_1 (24) will allow 12 units of X_1 to be produced, the available supply of S_2 (three) will allow 6 units of X_1 to be produced, and the available supply of S_3 (five) will allow 5 units of X_1 to be produced. The limiting resource then is S_3, so the S_3 row becomes the pivot row. The element at the intersection of the pivot column and pivot row is the "pivot element." This determination of the pivot row can be done quickly by generating what will be called "Q-values." This is done for each potential pivot row by dividing the resource value (in the last column) by the element in the pivot column, and registering this quotient in a "Q-column" to the right of the table. After all Q's have been determined, the row with the *smallest absolute value* of Q becomes the pivot row.

---

[1] The algebraic logic of this is also covered in Appendix B.

[2] This is also true if the problem is one of minimization, as will be explained on page 7.

Potential Pivot Element. For reasons explained in Appendix B, *any potential pivot element must be negative* and the *resource value in any potential pivot row must be positive;* therefore, all the Q-values that are of concern will be negative.[1]

The *pivot column* contains the largest bottom positive element and a potential pivot element (negative).
The *pivot row* generates the smallest absolute value of Q.

	X_1	X_2	1(Po)	Q	
S_1 =	—2	— 3	24	—12	
S_2 =	—.5	—.25	3	— 6	
S_3 =	—1*	0	5	— 5	← pivot row
Z =	3	2	0		

↑
Pivot column

*pivot element (negative)

Figure 1: Beginning matrix of example problem.

Rules for the Pivoting Procedure

The rules for the pivoting procedure, which are derived in Appendix B, are as follows:

A new matrix is developed by:

Pivoting Rules

PR-1. Changing the old pivot element to its reciprocal value.

PR-2. Changing the other values in the pivot column by dividing each by the old pivot element.

PR-3. Changing the other values in the pivot row by dividing each by the old pivot element *and giving it the opposite sign.*

PR-4. Changing all other elements in the matrix by subtracting from each: the value obtained by multiplying the element in the same column and pivot row by the element in the same row and pivot column, then dividing this product by the pivot element. ("Rectangle Rule")

[1] At this point, we are not concerned with situations in which negative Q-values *cannot* be generated. This will be covered later, see bottom of page 12.

This last rule may seem rather complicated, but it will prove simple if stepped through slowly. It may be thought of in terms of a rectangle. The elements involved form a perfect rectangle within the matrix. The element being changed is one corner of the rectangle, the pivot element is the opposite corner, and the two elements that are to be multiplied together complete the rectangle. In the example, the S_1 resource (24) becomes 24, minus the value of 5 times minus 2 divided by minus 1, or,

$$24 - [(5)\ (-2)/(-1)] = 24 - (+10) = 14$$

	X_1	X_2	1(Po)	Q
$S_1 =$	—2	— 3	24	—12
$S_2 =$	—.5	—.25	3	— 6
$S_3 =$	—1*	0	5	— 5 ←
$Z =$	3	2	0	

↑

Old Matrix

	S_3	X_2	1(Po)	Q	
$S_1 =$	2	— 3	14	—4⅔	
$S_2 =$	.5	—.25*	.5	—2	← new pivot row
$X_1 =$	—1	0	5		
$Z =$	—3	2	15		

↑ new pivot column

New Matrix

It will be noticed that in the new matrix, the headings "X_1" and "S_3" have been switched, which can be visualized by the fact that they have been "pivoted" on the element in the matrix common to both — the pivot element.

This new matrix indicates that the objective solution has been increased to 15 by producing five units of X_1. There are 14 units of S_1 and .5 units of S_2 left over. The positive 2 in the bottom row, however, indicates that the objective solution can be made even greater. A third matrix is developed using the X_2 column as the pivot column and the S_2 row as the pivot row.

New matrices will continue to be developed as long as there is a positive element in the bottom row (excepting the objective solution, of course). The solution of this example problem comes with the fourth matrix.

	S_1	S_2	1(Po)
$S_3 =$	—.25	3	2
$X_2 =$	—.5	2	6
$X_1 =$	.25	—3	3
$Z =$	—.25	—5	21

Solution

This final matrix presents all the standard information obtainable from the more common simplex methods; objective solution, activity levels, amounts of excess resources, shadow prices, and coefficients from which other information, such as variable prices and resource programming can be calculated:

Activity levels:	6 units of X_2
	3 units of X_1
Objective solution:	21
Excess resources:	2 units S_3 unused
Shadow prices:	S_1 price is .25
	S_2 price is 5

Minimizing Problems and Problems Where Zero Production Is Not Allowed

The preceding example was a case where zero production was allowed under the given constraints, and the objective function was to be maximized. A common L.P. problem is one where zero production is not allowed by the constraints and where the objective function is to be minimized.

Minimizing a positive objective function is nothing more than maximizing a negative objective function. If an objective function, $Z = 8X_1 + 4X_2$, is to be minimized, it is the same as maximizing $-Z = -8X_1 - 4X_2$. In this case the bottom row of the beginning matrix would contain:

	(X_1)	(X_2)	$(1(Po))$
$-Z =$	-8	-4	0

,

indicating that the objective solution could be reduced by the amounts shown for X_1 and X_2. Reducing by a negative amount, however, means adding to or increasing the objective solution, which would defeat the minimizing objective. This objective solution can be reduced further only if a positive value appears in the bottom row, which leaves the previously stated rule (see page 3, *Pivot Column*) unchanged. The objective solution will naturally be a negative, but with a simple sign change the objective solution becomes an acceptable value. For example, if $-Z = -85$, then $Z = 85$.

This minimizing case is directly related to the case where zero activity is not allowed. The first case requires the second, for minimizing a problem where zero activity is allowed leads to the trivial solution of zero, no activity.

In the example problem, suppose that the second constraint were changed to:

$$.5X + .25X_2 \geqq 3$$

(greater than, or equal to). In order to make this into an equality a slack variable must be *subtracted* from the left hand side of the inequality sign: $.5X_1 + .25X_2 - S_2 = 3$. When this equality is restated in terms of the positive slack variable, a negative sign appears in the last term:

$$S_2 = .5X_1 + .25X_2 - 3.$$

If this modified problem were now to be minimized, the initial matrix would be:

	X_1	X_2	1(Po)
$S_1 =$	−2	−3	24
$S_2 =$	.5	.25	−3
$S_3 =$	−1	0	5
$-Z =$	−3	−2	0

The bottom row indicates that the problem has been solved because of the negative numbers, however, the matrix says this is accomplished by having a negative amount of S_2 in solution, which is not an acceptable situation. The negative three in the last column must be removed. It is here that this approach differs from the more commonly presented techniques. Rather than becoming involved with an artificial variable, its high negative value and additional computer storage space, the negative last column figure is removed mathematically with the following rules:

Rules for Removing Negative Resources[1]

Neg–1. The row containing the negative number in the last column becomes the pivot row.

Neg–2. The pivot element must be *positive.*

Neg–3. The pivot column is chosen arbitrarily if more than one column satisfies rules Neg–1 and Neg–2.

Neg–4. If there are two or more negative numbers in the last column, then the pivot row is the row generating the *largest* absolute value of Q.

Once the pivot row and pivot column are determined, the pivoting rules are followed as before. The first iteration for the above example is shown here:

	X_1	X_2	1(Po)
$S_1 =$	−2	−3	24
$S_2 =$	.5*	.25	−3
$S_3 =$	−1	0	5
$-Z =$	−3	−2	0

Old Matrix

	S_2	X_2	1(Po)
$S_1 =$	−4	−2	12
$X_1 =$	2	−.5	6
$S_3 =$	−2	.5*	−1
$-Z =$	−6	−.5	−18

New Matrix

[1] These rules are derived in Appendix B.

In the new matrix the negative three has been eliminated, however, a negative one has appeared in the S_3 row. For the next pivoting process, the S_3 row will therefore become the pivot row, and the X_2 column will become the pivot column since it contains the only positive element in the S_3 row. The third and final table,

	S_2	S_3	1(Po)
S_1 =	−12	−4	8
X_1 =	0	−1	5
X_2 =	4	2	2
−Z =	− 8	−1	−19

indicates that the solution is minimized at 19 (− Z = − 19, therefore Z = 19). All negative values in the last column have been eliminated and the negative numbers in the bottom row indicate that no further minimizing can be accomplished. If, after eliminating all the last column negative values, positive numbers exist in the bottom row, the matrix is further iterated by choosing the pivot column and the pivot row in the manner described on page 3.

Summary of Rules for Determining Pivot Rows and Pivot Columns and the Pivoting Operation

Once the constraints for an L. P. problem are set up in inequality form, they are made into equalities by adding or subtracting slack variables. These equality constraints are then restated in terms of the *positive* slack variable, and set up in the following matrix form:

	X_1	X_2	. . .	1(Po)
$S_1 =$				
$S_2 =$				
.				
.				
.				
$Z =$				0

To the bottom row of this matrix, the objective function is added. If the objective function is to be maximized, it takes the form: $Z = X_1 + X_2 \ldots X_n$. If it is to be minimized, it takes the form: $-Z = -X_1 - X_2 - \ldots - X_n$. The objective solution is initially zero.

I. If any negative figures appear in the last column:

Rules for Removing Negative Resources

Neg–1. The row in which the negative figure appears is the pivot row.

Neg–2. The pivot element must be positive.

Neg–3. The pivot column is arbitrarily chosen.

Neg–4. If there are two or more negative numbers in the last column, then, using the arbitrarily chosen pivot column, Q-values are generated for each row containing a negative number. The row with the *largest* absolute value of Q will be the pivot row (all Q's will be negative since the potential pivot element must be positive and since only the negative last row numbers are of concern).

II. Once the pivot row and pivot column are determined, the pivoting process develops a new matrix by:

Pivoting Rules

PR–1. Changing the old pivot element to its reciprocal value.

PR–2. Changing all other elements in the pivot column by dividing them by the old pivot element.

PR–3. Changing all other elements in the pivot row by dividing them by the old pivot element and giving them the opposite sign.

PR–4. Changing all other elements by subtracting from them the value derived from the "rectangle rule" (see PR–4, page 4).

This process of determining the pivot row and column and pivoting the matrix is continued until all negative values are eliminated from the last column.

III. When this is accomplished, or when the initial matrix does not contain negative values in the last column, and when positive values are contained in the bottom row (excepting the solution box), the pivot row and column for successive pivoting processes are determined as follows:

Rules for Determining Pivot Columns and Pivot Rows

Pivot Column: The column with the largest positive bottom element and which contains a negative potential pivot element becomes the pivot column.

Pivot Row: Q-values are generated for all rows where the potential pivot element is negative. The smallest absolute value of Q determines the pivot row (it is obvious that again all values of Q will be negative).

Once the pivot row and column have been determined, the pivoting process continues as previously described (see II, page 10).

This method (III) of determining the pivot row and column, and the pivoting process (II) are continued until only negative values are contained in the bottom row. The problem is then solved.

Equality Constraints and No Solution Situations

There are two items that warrant brief coverage due to their special handling in the specific technique described herein.

The Equality Constraint

Because this technique does not use artificial variables as do the more conventional methods, it is recommended that the equality constraint be handled as two opposite inequality constraints, adding and subtracting slack variables where indicated. Handled in this manner, one of the two slack variables will equal zero in the final solution.

It is also possible to use one of the unknowns as if it were a slack variable. In the constraint $X_1 + X_2 = 100$, for example, X_1 or X_2 could be used to absorb the "unused" portion of the 100, thus developing the constraint: $X_1 = -X_2 + 100$.

Now that X_1 has been "solved" (in terms of X_2), however, it becomes necessary to restate all other constraints (that contain X_1), and the objective function, in terms of X_2.

In our example problem, the constraint $2X_1 + 3X_2 \leq 24$, becomes $2(-X_2 + 100) + 3X_2 \leq 24$, and the representative equality becomes: $S_1 = -X_2 - 176$. The objective function, $Z = 3X_1 + 2X_2$ max.) becomes $X = 3(-X_2 + 100) + 2X_2$, or $Z = -X_2 + 300$, and the initial solution is no longer zero but 300.

An equality constraint handled in this manner reduces the number of constraints by one, and reduces the number of iterations by two, but, as is quite evident, requires a great deal more time in setting up the problem.

This increased "set up" time and the increased probability of making simple mathematical errors while restating the constraints and objective function, appear to the authors as justification for handling an equality constraint in the first described manner. Therefore, unless the cost of the additional "solving time" is prohibitive, or the additional required constraint causes the problem to become too unmanageable, it is recommended that an equality constraint be handled as two opposite inequality constraints.

The "No Solution" Situation

The two most common "no solution" situations that occur in an L. P. problem are: (1) when a function is to be maximized, but there is no upper-limit constraint ("unbounded solution"), and (2) when

two or more constraints contradict each other ("mathematical inconsistency").

When either of these situations occur following this technique, definite and unique events will stop the iteration process. If a maximum objective is called for when no upper limit is set by the constraints, the matrix-iteration process will run into a dead end by having a positive value in the bottom row, indicating that the solution can be made larger, and positive values in the last column, *but there will be no negative elements to qualify as a pivot element* (negative Q-values cannot be generated).

If two or more constraints contradict each other, the pivoting process will dead-end because a negative value will appear in the last column, *but there will be no positive element to qualify as a pivot element* (again, no negative Q-values can be generated).

Thus, tests for "boundedness" and mathematical consistency are built into this technique. In the accompanying computer program (Appendix A), error messages to this effect are included.

Computer Program

A Fortran II program for this L. P. problem solving technique is presented in Appendix A.

It is arbitrarily dimensioned for a 14 x 14 matrix which would be the maximum size problem that could be run in a 20,000 unit storage capacity computer if numerical tables and/or other management routines have reduced the storage area to approximately 8,000 units.

The output gives the initial matrix, the activities being pivoted and the objective function for each iteration, the final solution activity levels, the final objective function (solution) and the shadow prices.

Selected References

(1) Baumol, William J., *Economic Theory and Operations Analysis.* (Englewood Cliffs; Prentice-Hall Inc., 1961).

(2) Heady, Earl O. and Wilfred Candler, *Linear Programming Methods.* (Ames: The Iowa State University Press, 1963).

(3) Stiefel, Eduard L., *An Introduction to Numerical Mathematics.* (New York: Academic Press, 1963).

(4) Vajda, S., *An Introduction to Linear Programming and the Theory of Games.* (New York: John Wiley & Sons, Inc., 1960).

(5) Vajda, S., *Mathematical Programming.* (Reading, Mass.: Addison-Wesley Publishing Company, Inc., 1961).

APPENDIX A

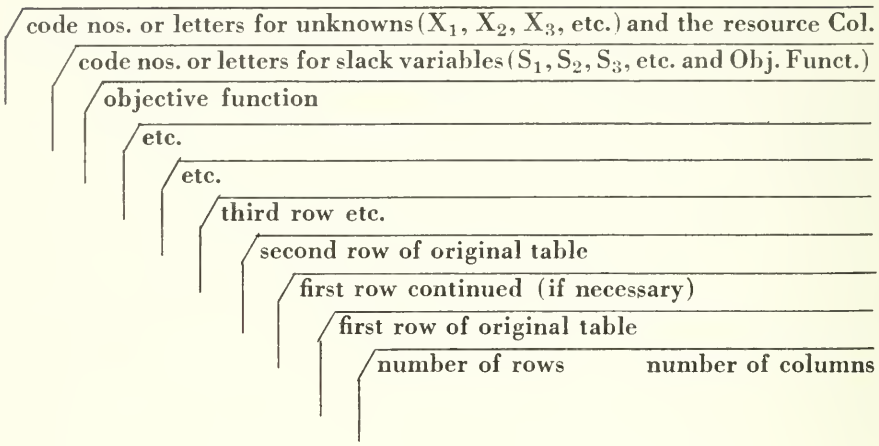
Fortran II
Computer Program

```
      COMMENT- THIS PORTION OF THE PROGRAM READS IN
      THE PROBLEM DATA AND PRINTS IT OUT IN TABLE FORM.
   DIMENSION A(14,14), KR(14), KC(14)
30 FORMAT (16I5)
40 FORMAT (1H ,39H NO SOLUTION, CONTRADICTING CONSTRAINTS///)
50 FORMAT (1H ,32H NO SOLUTION, INFINITE OBJECTIVE///)
60 FORMAT (1H ,I5,7F10.4)
70 FORMAT (1H ,10X,2I10,5X,F10.4)
80 FORMAT (1H0,5X,7I10)
90 FORMAT (8F10.4)
   READ  30, M, N
   DO 1 I = 1,M
 1 READ  90,(A(I,J), J = 1,N)
   READ  30,(KR(I), I = 1,M)
   READ  30,(KC(J), J = 1,N)
   PRINT  80, (KC(J), J = 1,N)
   DO 2 I = 1,M
 2 PRINT  60, KR(I),(A(I,J),J = 1,N,
   MI = M-1
   NI = N-1
      COMMENT- THE FOLLOWING STEPS ASK THE QUESTION,
      IS THERE A NEGATIVE VALUE IN THE LAST COLUMN.
 3 DO 4 I = 1,MI
   IF (A(I,N)) 5, 4, 4
 4 CONTINUE
   GO TO 11
      COMMENT- IF THE ANSWER IS YES, THE FOLLOWING STEPS
      DETERMINE THE APPROPRIATE PIVOT ROW AND COLUMN.
 5 Q = 0.
   DO  10 J = 1,NI
   DO 9 I = 1,MI
   IF (A(I,J)) 9, 9, 6
 6 IF (A(I,N)) 7, 9, 9
 7 IF (Q - A(I,N)/A(I,J)) 9, 9, 8
 8 Q = A(I,N)/A(I,J)
   KROW = I
   KCOL = J
 9 CONTINUE
   IF (Q) 20, 10, 20
10 CONTINUE
   PRINT 40
   GO TO 95
      COMMENT- IF THE ANSWER IS NO, THE FOLLOWING STEPS
      ASK THE QUESTION, IS THERE A POSITIVE VALUE IN
      THE BOTTOM ROW.
11 QT = 0.
   DO 14 J =1,NI
   IF (A(M,J)) 14, 14, 12
12 IF (QT - A(M,J)) 13, 14, 14
13 QT = A(M,J)
   KCOL = J
14 CONTINUE
   IF (QT) 95, 95, 15
      COMMENT- IF THE ANSWER IS NO, THE PROBLEM IS SOLVED
      AND THE SOLUTION IS PRINTED OUT.
      IF THE ANSWER IS YES, THE FOLLOWING STEPS DETERMINE
      THE APPROPRIATE PIVOT ROW AND COLUMN.
15 Q= -99999999.
   DO 18 I = 1,MI
   IF (A(I,KCOL)) 16, 18, 18
```

```
16 IF (Q - A(I,N)/A(I,KCOL)) 17, 18, 18
17 Q = (A(I,N)/A(I,KCOL))
   KROW = I
18 CONTINUE
   IF (Q + 99999999.) 20,  19, 20
19 PRINT 50
   GO TO 95
        COMMENT- ONCE THE APPROPRIATE PIVOT ROW AND COLUMN
        ARE DETERMINED THE FOLLOWING STEPS ARE THE
        PIVOTING PROCESS. THE OPERATION THEN RETURNS
        TO THE FIRST QUESTION (STEP NO. 3).
20 KEEP  = KR(KROW)
   KR(KROW) = KC(KCOL)
   KC(KCOL) = KEEP .
   DO 23 II= 1,M
   IF (II- KROW) 21, 23, 21
21 DO 23 JJ= 1,N
   IF (JJ- KCOL) 22, 23, 22
22 A(II,JJ) = A(II,JJ) - (A(KROW,JJ)*A(II,KCOL))/A(KROW,KCOL)
23 CONTINUE
   DO 25 II= 1,M
   IF (II- KROW) 24, 25, 24
24 A(II,KCOL) = A(II,KCOL)/A(KROW,KCOL)
25 CONTINUE
   DO  27 JJ= 1,N
   IF (JJ- KCOL) 26, 27, 26
26 A(KROW,JJ) = (-1.)*(A(KROW,JJ)/A(KROW,KCOL))
27 CONTINUE
   A(KROW,KCOL) = 1./A(KROW,KCOL)
   PRINT 70, KR(KROW),KC(KCOL),A(M,N)
   GO TO 3
        COMMENT- THE FOLLOWING STEPS PRINT OUT THE FINAL
        SOLUTION AND END THE OPERATION.
95 DO 96 I = 1,M
96 PRINT 60 , KR(I),A(I,N)
   DO 97 J = 1,NI
97 PRINT 60 , KC(J),A(M,J)
   END
```

Data Cards

The preceding program calls for data cards to be arranged in the following order; each punched according to the indicated format.


code nos. or letters for unknowns (X_1, X_2, X_3, etc.) and the resource Col.
code nos. or letters for slack variables (S_1, S_2, S_3, etc. and Obj. Funct.)
objective function
etc.
etc.
third row etc.
second row of original table
first row continued (if necessary)
first row of original table
number of rows
number of columns

APPENDIX B

Algorithm Rules Derivation

Appendix B

Derivation of Pivoting Rules and Pivot Row and Column Selection Rules

I. Pivoting Rules

The pivoting process is a method of solving simultaneous equations. By algebraically solving two such equations and following the steps with the coefficient matrix, rules for the pivoting process are developed.

Given: two general equations, and the corresponding coefficient matrix:

$S_1 = aX_1 + bX_2 + k_1$

$S_2 = cX_1 + dX_2 + k_2$

	X_1	X_2	1
$S_1 =$	a	b	k_1
$S_2 =$	c	d	k_2

The first step is to solve for X_1 and S_2 (in terms of X_2, S_1, a, b, c, d, k_1 and k_2). The column headed "X_1" is the *pivot column.* The row headed "S_1" is the *pivot row.* The element at the intersection is the *pivot element.*

Solving for X_1:

$$aX_1 = S_1 - bX_2 - k_1$$

$$X_1 = 1/a(S_1) - b/a(X_2) - k_1/a \qquad \text{(Equation 1)}$$

Solving for S_2:

$$S_2 = c[1/a(S_1) - b/a(X_2) - k_1/a] + dX_2 + k_2$$

$$S_2 = c/a(S_1) - c(b/a)(X_2) - c(k_1/a) + dX_2 + k_2$$

Combining terms —

$$S_2 = c/a(S_1) + [d - c(b/a)]\,X_2 + [k_2 - c(k_1/a)] \qquad \text{(Equation 2)}$$

Equations (1) and (2) yield the matrix:

	S_1	X_2	1
$X_1 =$	1/a	—b/a	—k_1/a
$S_2 =$	c/a	d — c(b/a)	k_2 — c(k_1/a)

From this, four rules can be drawn for the pivoting process:

PR–1. The pivot element (a) becomes its reciprocal value (1/a).

PR–2. The other elements in the pivot column (c) become themselves divided by the pivot element (c/a).

PR–3. The other elements in the pivot row (b and k_1) become themselves divided by the pivot element and given the opposite sign ($-$ b/a and $-$ k_1/a).

PR–4. Each element in the remaining part of the matrix (d and k_2 becomes itself less the quantity derived by multiplying the element in the same row and in the pivot column by the element in the same column and in the pivot row, and dividing this product by the pivot element: d $-$ c(b/a) and $k_2 - c(k_1/a)$.

These rules may be checked by completing the solution, that is, exchanging S_2 and X_2, thus solving for the X's in terms of the S's.

II. Determining Pivot Rows and Pivot Columns

Given: the generalized coefficient matrix:

	X_1	X_2	X_3	. . .	X_q	. . .	X_n	1(Po)
$S_1 =$	a	a	a	. . .				C_1
$S_2 =$	a	a	. . .					C_2
.	.	.						.
.	.	.						.
.	.	.						.
$S_p =$					a_{pq}			C_p
$S_k =$					a_{kq}			C_k
.								.
.								.
.								.
$S_m =$							a_{mn}	C_m
$Z =$	a_1	a_2	. . .		a_q	. . .	a_n	C*

(Initially C* = 0)

In this generalized matrix, S_p is our eventual pivot row and X_q is our eventual pivot column (i.e. a_{pq} is our eventual pivot element).

There are two requirements on which we will insist:

A. Only solutions that are acceptable values will be considered ($X_1 \geqq 0, X_2 \geqq 0, \ldots X_n \geqq 0, S_1 \geqq 0, S_2 \geqq 0, \ldots S_m \geqq 0$), therefore, the C's (last column values) must remain or become $\geqq 0$.

B. During each exchange, C* (the objective solution) must increase (at least not decrease).

IF ALL C'S ARE NON-NEGATIVE

From the above requirements and the pivoting rules *(assuming all C's are non-negative)*, three conclusions can be drawn:

1–1. Since $C_p \rightarrow$ (becomes) $- C_p$/pivot (pivoting rule #3), which must remain $\geqq 0$, we must choose a pivot element that is negative.

1–2. Since $C^* \rightarrow C^* - (C_p a_q)$/pivot (pivoting rule #4), which must be non-decreasing, we must choose a pivot column so that the quantity $(C_p a_q)$/pivot is $\leqq 0$, i.e. a_q must be positive.

1–3. Since $C_k \rightarrow C_k - (C_p a_{kq})$/pivot (pivoting rule #4) which must remain $\geqq 0$, we must choose a pivot row so that $C_k \geqq (C_p a_{kq})$/pivot.

Conclusion 1–3 is automatically satisfied if $a_{kq} \geqq 0$.

However, if there is more than one a_{iq} in the pivot column which is negative, we will gain by being more cautious.

Suppose a_{kq} is negative, then Conclusion 1-3 can be written:

$$C_k / a_{kq} \leqq C_p \text{/pivot, or } C_k / a_{kq} \leqq C_p / a_{pq} .$$

If we call C_k / a_{kq}, "Q_k" (the kth "characteristic quotient"), then the gain comes by choosing the pivot row so that Q_p is the largest of the Q's, for this will prevent any of the positive C's from becoming negative. Since the Q's in which we are interested are all negative, we can conclude that our pivot row is the one which has the smallest absolute value of Q.

From these three conclusions, three rules can be made for determining the appropriate pivot row and pivot column for the pivoting process if all C's are non-negative.

Pivot element: The pivot element must be negative.

Pivot column: The bottom element in the pivot column must be positive.

Pivot row: Among rows with negative a_{iq}'s, the pivot row is the row which has the smallest absolute value of Q.

IF SOME C'S ARE NEGATIVE

A linear programming problem that contains greater-than-or-equal-to constraints will have negative values appearing in the last column of the initial table. These negative values must *become* non-negative (see requirement A, page 22). Given this situation and considering the pivoting rules, three conclusions can be drawn:

2–1. Since we are only concerned with the negative C's, only rows with negative C's will be considered for the pivot row.

2–2. Since $C_p \rightarrow -C_p/\text{pivot}$, we must choose a positive pivot element (i.e. $a_{pq} > 0$).

2–3. If there is only one row that contains a negative C and at least one positive element in that row, there is no problem of determining the appropriate pivot row and pivot column for the pivoting process (the choice of a pivot column is arbatrary). However, if there are two or more rows containing negative C's, we will gain by being cautious.

Suppose that C_k is negative and a_{kq} is positive (see generalized table page 21).

Since $C_k \rightarrow C_k - (C_p a_{kq})/\text{pivot}$, it would be desirable if $(C_p a_{kq})/\text{pivot}$ were $\leq C_k$ since this would cause C_k to become non-negative also. This can be modified to read: $C_p/\text{pivot} \leq C_k/a_{kq}$ or $C_p/a_{kq} \leq C_k/a_{kq}$ or even further $Q_p \leq Q_k$. Since again the Q values we are dealing with are all negative, we can conclude that we would gain if the pivot row is one which has the largest absolute value of Q.[1]

From these three conclusions (2–1, 2–2, and 2–3), four rules can be made for determining the appropriate pivot row and column for the pivoting process if some C's are negative.

---

[1] It is possible that a previously non-negative C may become negative during a pivoting process. This is of no concern, however, since it will eventually be eliminated in the same manner as the other negative C's.

Neg–1. Only rows with negative C's will be eligible for the pivot row.

Neg–2. The pivot element must be positive.

Neg–3. The pivot row is the row which has the largest absolute value of Q.

Neg–4. The choice of a pivot column is arbitrary, provided the first three rules are satisfied.